Informatics Institute of Technology

Foundation Certificate programme

Individual Course Work Assignment

I’m export with this my current data base.

Student Name : Thilini Abeywickrama

Student ID : 20200476

Unit Code : DOC334

Module : Introduction to Programming II

Module Leader : Mr. Sudarshana Welihinda

Abstract

This report is a simple Hangman game using the Python programming language. We can use this as a small project to boost their programming skills and understanding of logic. The Hangman program randomly selects a secret word from a list of secret words. The random word will provide this ability, so line 1 in the program imports it. Hangman is a popular word game in which player chooses a secret word attempt to guess the word one letter at a time. If a guessed letter appears in the word, all instances of it are revealed. If not, the guesser loses a chance. If the player figures out the secret word, the Player wins. If not, the player will be defeated.

Acknowledgment

I have taken a lot of effort into this project. Primarily I want to thank God for giving me the courage and the strength to face all the challenges and complete this report. On the same accord, I would like to express my sincere gratitude to Mr. Nishan, Ms. Tharushi, and Ms. Keerthiga for their untiring support . I am thankful for them for directing me on the correct path and for providing the important statistics and sources that I needed for this report.

I would also like to thank all the other lecturers and non-academic staff of IIT.
Special thanks to my parents who have helped endlessly throughout this project.

The preparation of this report “Hangman” was an immense learning curve and it helped me to work on the python programming language.
Finally, I hope the experiences gained through this project will help me to develop my python programming skills as well.

Table of content

Contents

01. List of tables

02. List of figures

03. Problem Statement

You have to create a console Python 3.x program which will allow users to demonstrate the single player game called “Hangman”

Player guesses the correct word within the given number of turns. The result - Player wins.

Player runs out of turns before guessing the word. The result – Player loses. If the player guesses a character properly, his/her turn will not exhaust. Only the wrong guesses will exhaust turns. If the word consists of duplicate characters, a single guess will fill all the occurrence of that letter.

You have 20 stored words which will randomly appear during the game. All 20 words should have different number of turns based on the word size. As an addition you can display a small hint for the player about the word. Your program should also collect the player information and keep records of the games played. the collected information may have,

- Player’s name
- Word guessed
- Turns provided
- Turns used
- Win/lost status

05. Codes

5.1. Hangman game code

```
# to get random word
import random
#connect to the database
import Database

def print_game_status(remaining_guesses):
    mistakes ==  len(words)-1
    print("word: ",end='')

    for element in guesses:
        print(f"{element}",end='')
    print(f"\nYou have {remaining_guesses} turns remain")

# word list
words = ['elephant','apple','circle','square','rabbit','mouse','keyboard','fan','frock','frog','dog','cat','watermelon','chair','table','door','bag','rat','house','radio']
mistakes = 0
status = ""

# this is for choosing one word and count turns player have
def startgame():
    mistakes = 0
    status = ""
```

```
word_index = random.randint(0, len(words)-1)

remaining_guesses = len(words[word_index])

word = words[word_index].upper()


turns_provide = len(word)

for i in range(len(word)):

    guesses.append('_')


game_over = False


while not game_over:


    print_game_status(remaining_guesses)


          # inputs

    user_input = input("please enter a letter : ")


    if not user_input:

        print("That's not a valid input.please try again")

    else:

        letter = user_input[0].upper()

        if letter in word:

            for i in range(len(word)):

                if word[i] == letter:

                    guesses[i] = letter


            if '_' not in guesses:
                    game_over = True
            else:
                print("Sorry, that's not part of the word")
                remaining_guesses -= 1
                mistakes += 1
                if mistakes == len(words[word_index]):
                    game_over = True
         # win lost status
    if mistakes == len(words[word_index]):
        status = "Loss"
        print_game_status(remaining_guesses)
        print(f"Sorry,You lost. The word is : {word}")
    else:

        status = "Win"

        print("Congratulations......You won!")
        print(f"The word is : {word}")

    Database.insert_data(name,word,turns_provide,remaining_guesses,status)

 # inputs
name = input("Enter your name : ")
print("Hello", name + "!")
print("Let's play hangman......")
```

```
# This is for menu
while True:
    print()
    print("Welcome to Hangman")
    print(" ")
    print("__Menu__")
    print("")
    print(" A. Play ")
    print(" B. View History")
    print(" C. Exit")
    l = input("Enter your Menu Letter : ")
    if l == "A":
        guesses = []
        startgame()

    elif l == "B":
        Database.display_data()
    elif l == "C":
        print("Game End!")
        break
```

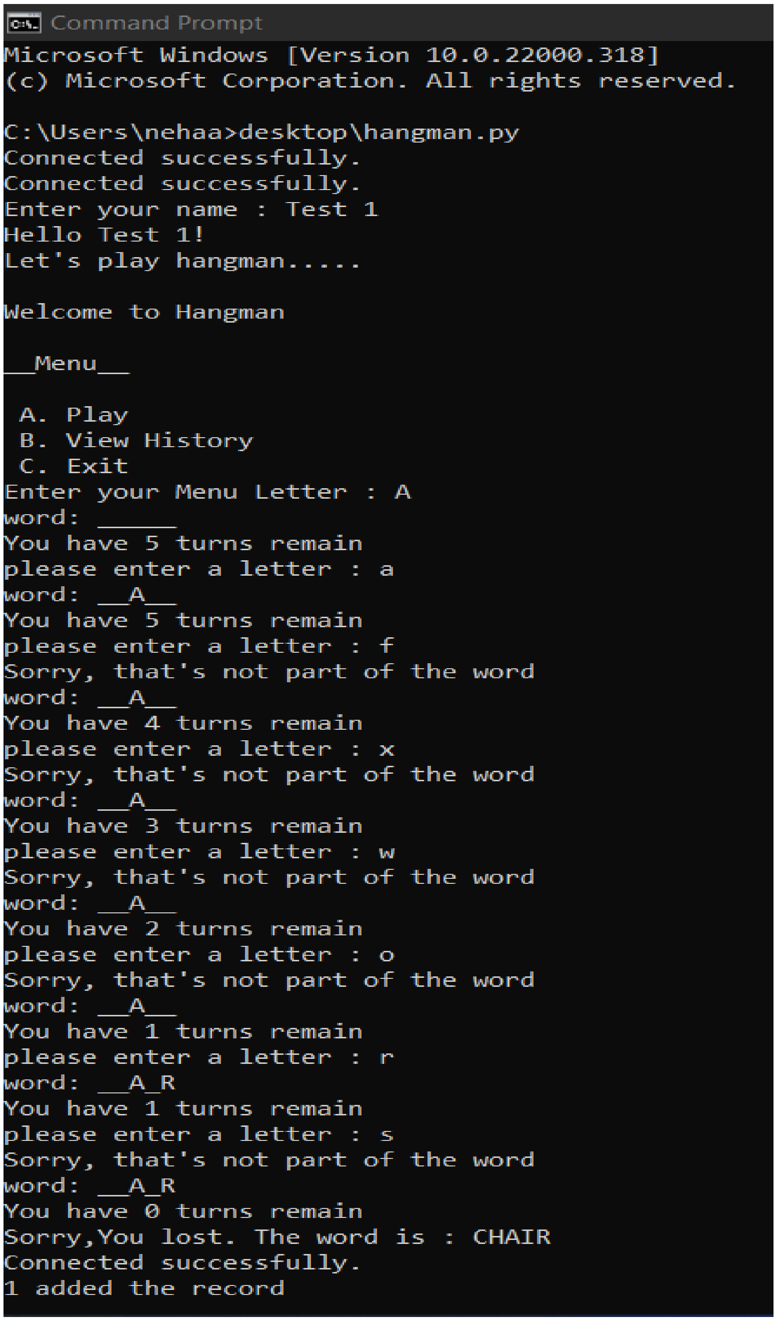
5.2. Hangman Database Code

```
# This is for check data connectivity
def hangman_db():
    import mysql.connector
    from mysql.connector import Error
    my_db = mysql.connector.connect(host="localhost",user="root", passwd="")
    try:
        if my_db.is_connected():
            print("Connected successfully.")
            new_cursor = my_db.cursor()
            new_cursor.execute("CREATE DATABASE IF NOT EXISTS Thilini_20200476;")
    except Error as e:
        print("Something went wrong", e)

#This function is creating a database name calling as thilini_20200476
def hangman_db_table():
    import mysql.connector
    from mysql.connector import Error
    my_table_conn = mysql.connector.connect(host="localhost", user="root",
passwd="",database="Thilini_20200476")
    try:
        if my_table_conn.is_connected():
            print("Connected successfully.")
            table_cursor = my_table_conn.cursor()
            table_cursor.execute("CREATE TABLE IF NOT EXISTS game_play(player_name
VARCHAR(50), word VARCHAR(25), turns_provided INT, turns_used INT,  status VARCHAR(15))")
```

```
except Error as e:
        print("Something went wrong", e)
def insert_data(name,word,turns_provide,turn_used,status):
    import mysql.connector
    from mysql.connector import Error
    add_data_conn = mysql.connector.connect(host="localhost", user="root", passwd="",
database="Thilini_20200476")
    try:
        if add_data_conn.is_connected():
            print("Connected successfully.")
            data_cursor = add_data_conn.cursor()
            add_command ="INSERT INTO game_play(player_name, word, turns_provided ,
turns_used  , status) VALUES (%s,%s,%s,%s,%s)"
            insert_params = (name,word,turns_provide,turn_used,status)
            data_cursor.execute(add_command, insert_params)
            print(data_cursor.rowcount,"added the record")
            add_data_conn.commit()
    except Error as e:
        print("Something is wrong", e)
def display_data():
    import mysql.connector
    add_data_conn = mysql.connector.connect(host="localhost", user="root", passwd="",
database="Thilini_20200476")
    #get all the records from the database
    print("\n----History----\n")
    data_cursor = add_data_conn.cursor()
    data_cursor.execute("SELECT * FROM game_play")
    myresult = data_cursor.fetchall()
    for x in myresult:
        for i in x:
            print(i,end="     ")
        print()
hangman_db()
hangman_db_table()
```

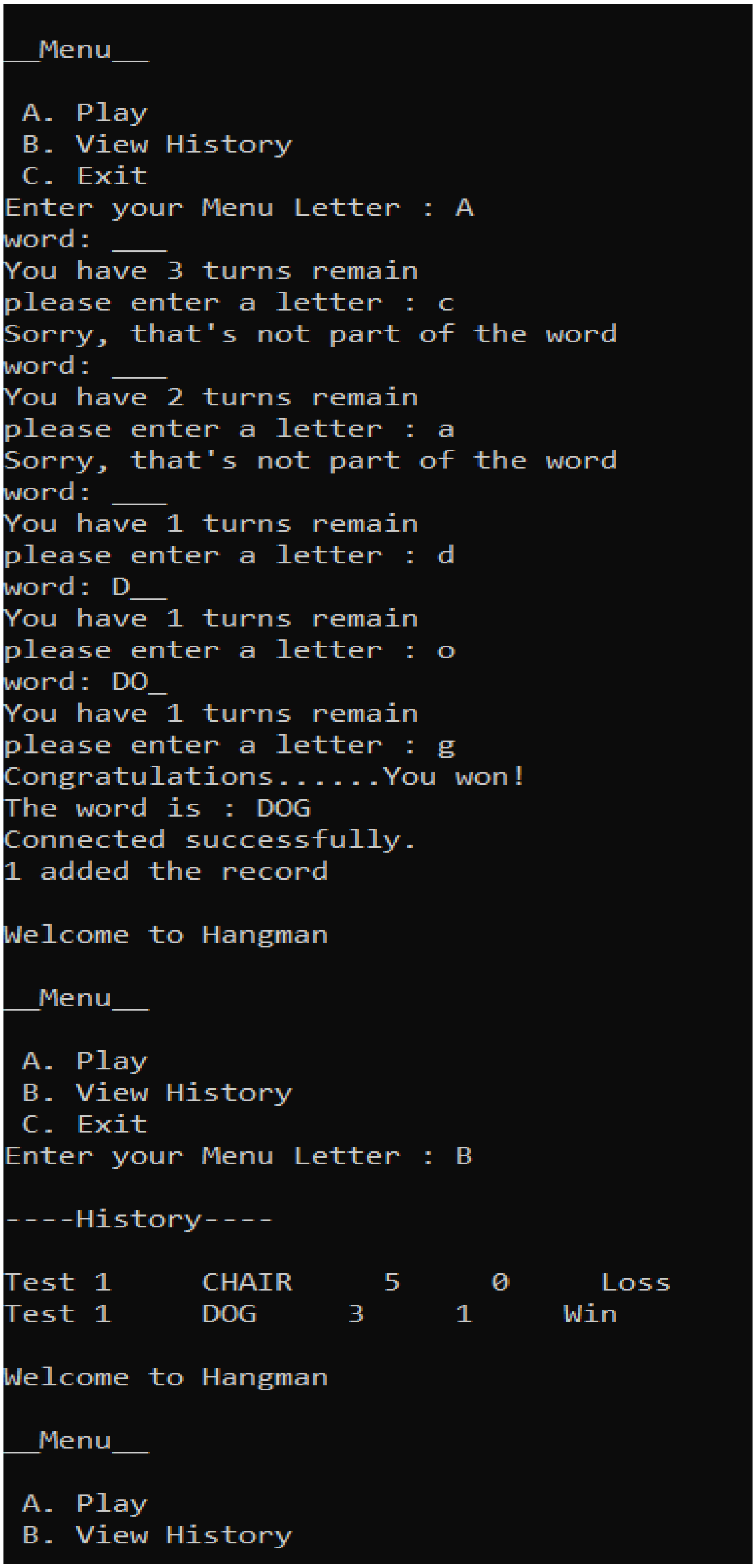
06. Test Cases

6.1 Test case 1



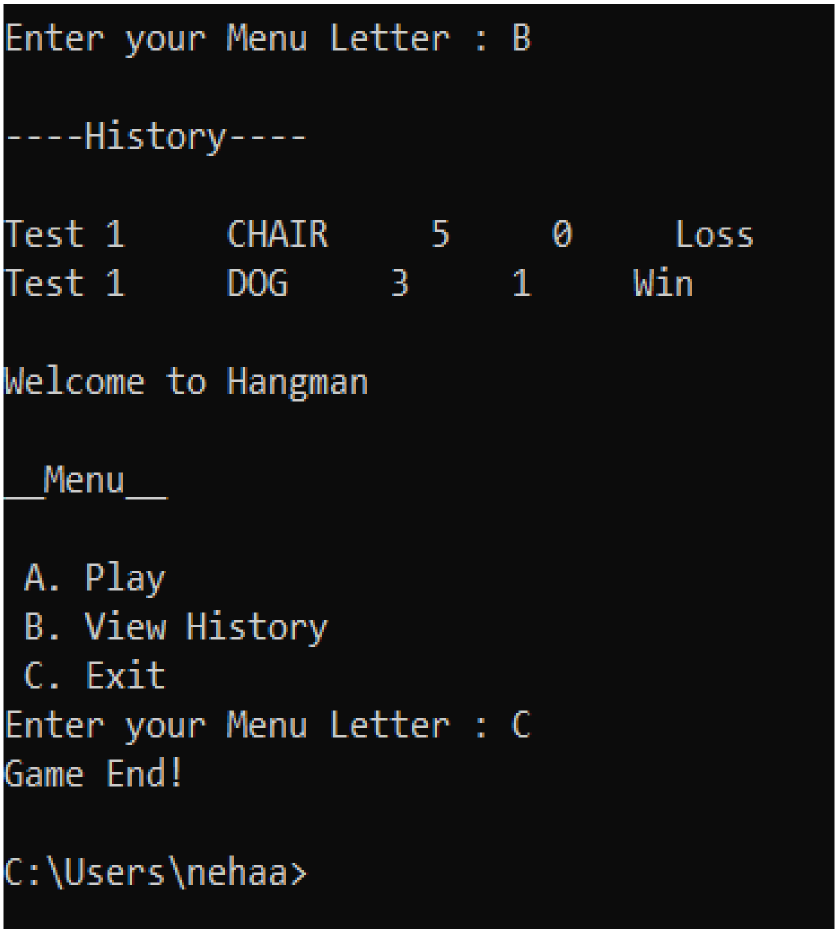
```
Microsoft Windows [Version 10.0.22000.318]
(c) Microsoft Corporation. All rights reserved.

C:\Users\nehaa>desktop\hangman.py
Connected successfully.
Connected successfully.
Enter your name : Test 1
Hello Test 1!
Let's play hangman.....

Welcome to Hangman

__Menu__

 A. Play
 B. View History
 C. Exit
Enter your Menu Letter : A
word: _____
You have 5 turns remain
please enter a letter : a
word: __A__
You have 5 turns remain
please enter a letter : f
Sorry, that's not part of the word
word: __A__
You have 4 turns remain
please enter a letter : x
Sorry, that's not part of the word
word: __A__
You have 3 turns remain
please enter a letter : w
Sorry, that's not part of the word
word: __A__
You have 2 turns remain
please enter a letter : o
Sorry, that's not part of the word
word: __A__
You have 1 turns remain
please enter a letter : r
word: __A_R
You have 1 turns remain
please enter a letter : s
Sorry, that's not part of the word
word: __A_R
You have 0 turns remain
Sorry,You lost. The word is : CHAIR
Connected successfully.
1 added the record
```

Figure 1



```
__Menu__

 A. Play
 B. View History
 C. Exit
Enter your Menu Letter : A
word: ___
You have 3 turns remain
please enter a letter : c
Sorry, that's not part of the word
word: ___
You have 2 turns remain
please enter a letter : a
Sorry, that's not part of the word
word: ___
You have 1 turns remain
please enter a letter : d
word: D__
You have 1 turns remain
please enter a letter : o
word: DO_
You have 1 turns remain
please enter a letter : g
Congratulations......You won!
The word is : DOG
Connected successfully.
1 added the record

Welcome to Hangman

__Menu__

 A. Play
 B. View History
 C. Exit
Enter your Menu Letter : B

----History----

Test 1     CHAIR     5     0     Loss
Test 1     DOG     3     1     Win

Welcome to Hangman

__Menu__

 A. Play
 B. View History
```

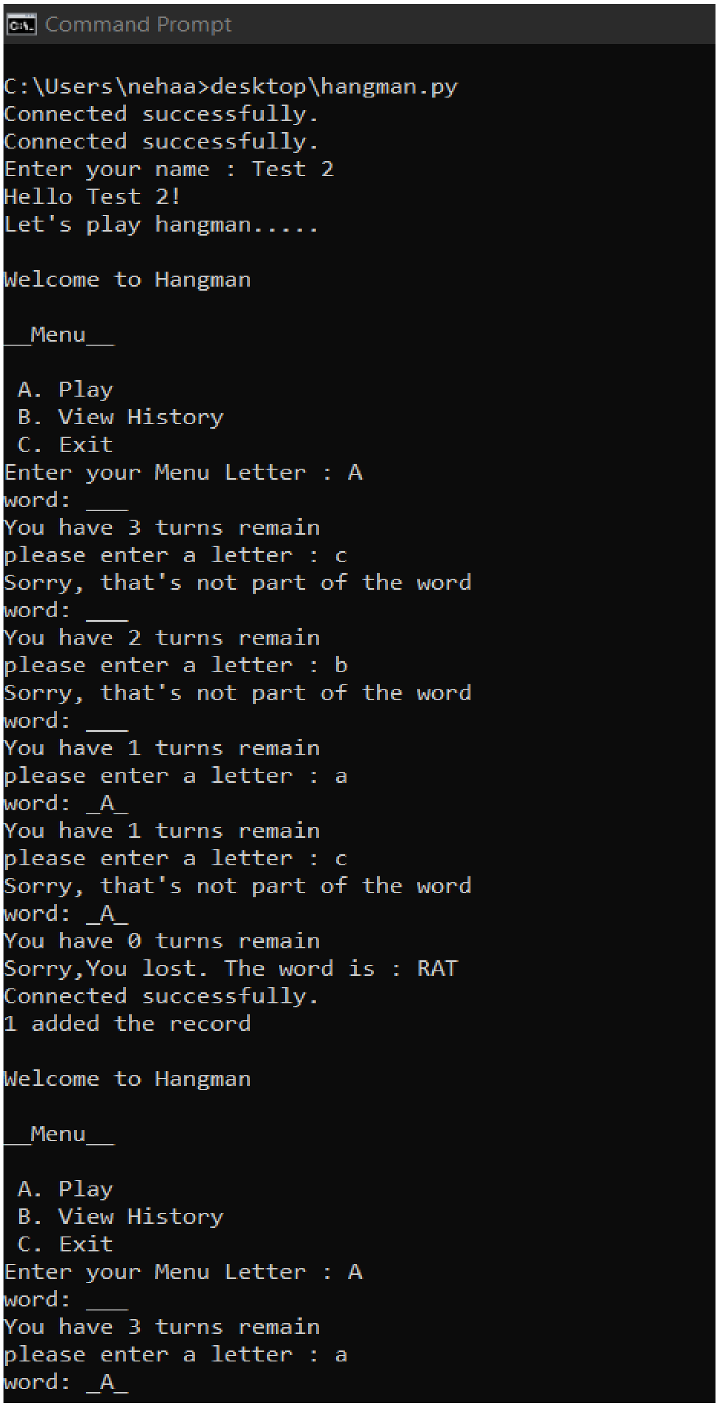
Figure 2



```
Enter your Menu Letter : B

----History----

Test 1     CHAIR     5     0     Loss
Test 1     DOG     3     1     Win

Welcome to Hangman

__Menu__

 A. Play
 B. View History
 C. Exit
Enter your Menu Letter : C
Game End!

C:\Users\nehaa>
```

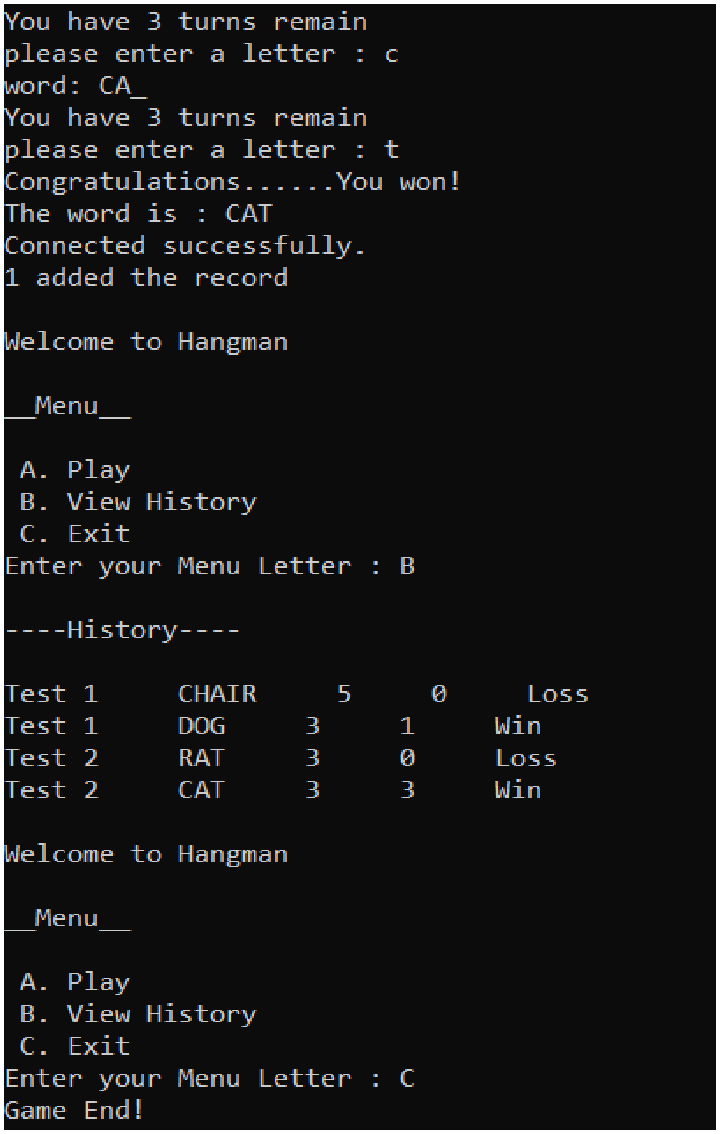
Figure 3

6.1 Test case 2



```
C:\Users\nehaa>desktop\hangman.py
Connected successfully.
Connected successfully.
Enter your name : Test 2
Hello Test 2!
Let's play hangman.....

Welcome to Hangman

__Menu__

 A. Play
 B. View History
 C. Exit
Enter your Menu Letter : A
word: ___
You have 3 turns remain
please enter a letter : c
Sorry, that's not part of the word
word: ___
You have 2 turns remain
please enter a letter : b
Sorry, that's not part of the word
word: ___
You have 1 turns remain
please enter a letter : a
word: _A_
You have 1 turns remain
please enter a letter : c
Sorry, that's not part of the word
word: _A_
You have 0 turns remain
Sorry,You lost. The word is : RAT
Connected successfully.
1 added the record

Welcome to Hangman

__Menu__

 A. Play
 B. View History
 C. Exit
Enter your Menu Letter : A
word: ___
You have 3 turns remain
please enter a letter : a
word: _A_
```

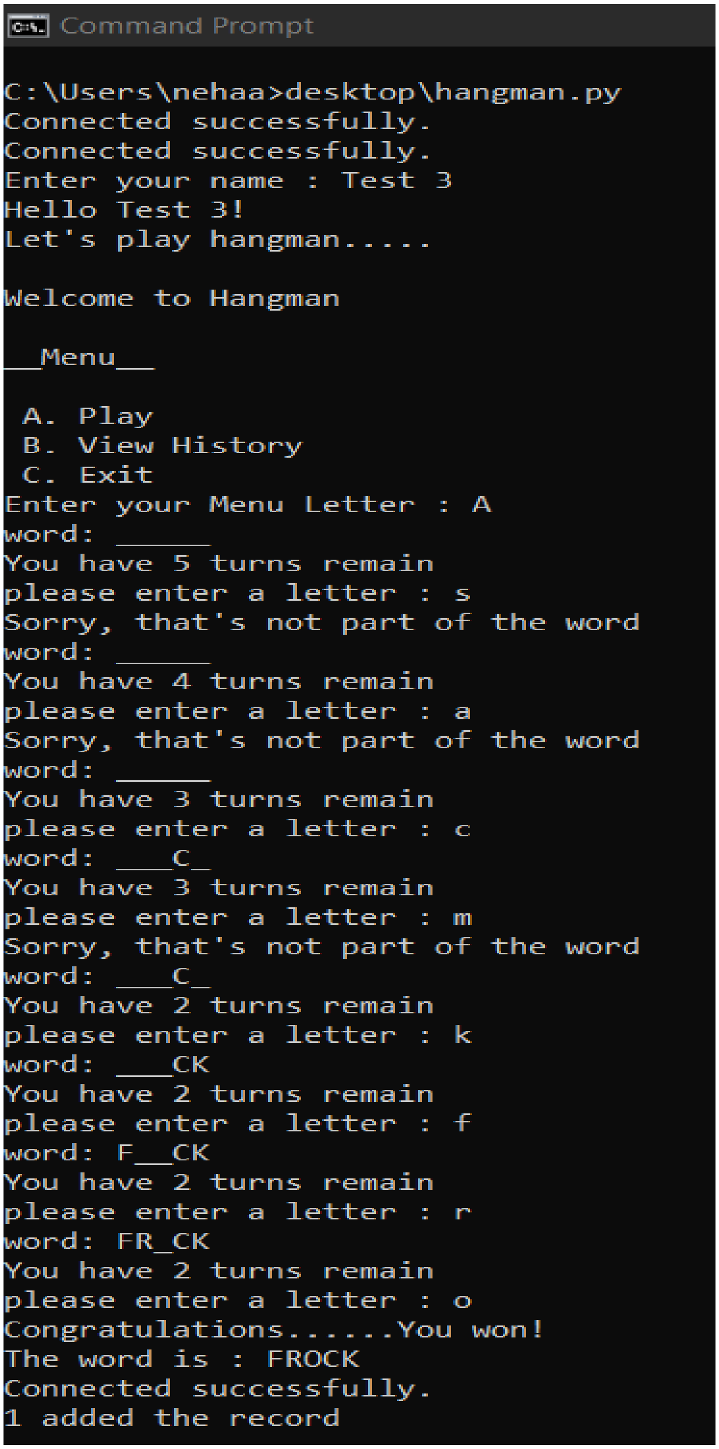
Figure 4



```
You have 3 turns remain
please enter a letter : c
word: CA_
You have 3 turns remain
please enter a letter : t
Congratulations......You won!
The word is : CAT
Connected successfully.
1 added the record

Welcome to Hangman

__Menu__

 A. Play
 B. View History
 C. Exit
Enter your Menu Letter : B

----History----

Test 1     CHAIR     5     0     Loss
Test 1     DOG     3     1     Win
Test 2     RAT     3     0     Loss
Test 2     CAT     3     3     Win

Welcome to Hangman

__Menu__

 A. Play
 B. View History
 C. Exit
Enter your Menu Letter : C
Game End!
```

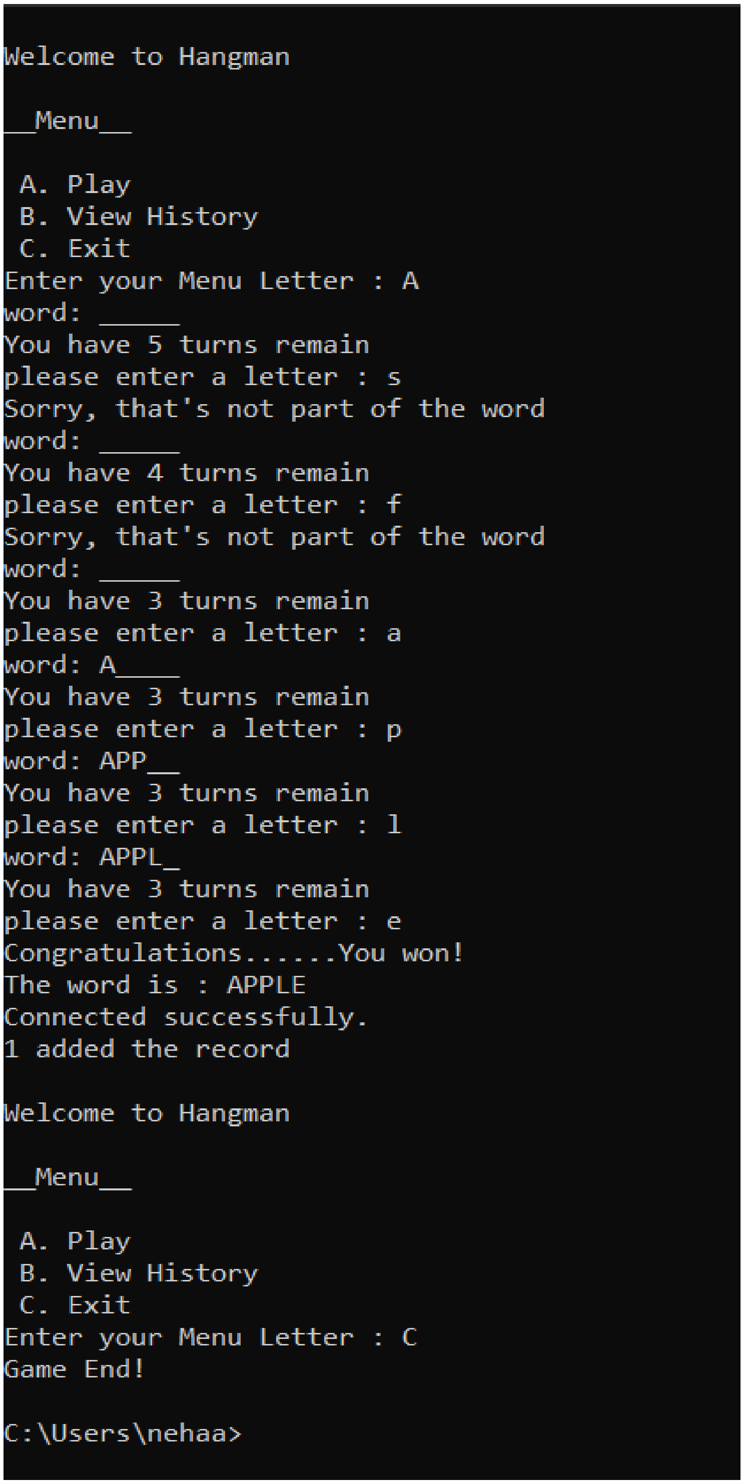
Figure 5

6.3 Test case 3



```
C:\Users\nehaa>desktop\hangman.py
Connected successfully.
Connected successfully.
Enter your name : Test 3
Hello Test 3!
Let's play hangman.....

Welcome to Hangman

__Menu__

 A. Play
 B. View History
 C. Exit
Enter your Menu Letter : A
word: _____
You have 5 turns remain
please enter a letter : s
Sorry, that's not part of the word
word: _____
You have 4 turns remain
please enter a letter : a
Sorry, that's not part of the word
word: _____
You have 3 turns remain
please enter a letter : c
word: ___C_
You have 3 turns remain
please enter a letter : m
Sorry, that's not part of the word
word: ___C_
You have 2 turns remain
please enter a letter : k
word: ___CK
You have 2 turns remain
please enter a letter : f
word: F__CK
You have 2 turns remain
please enter a letter : r
word: FR_CK
You have 2 turns remain
please enter a letter : o
Congratulations......You won!
The word is : FROCK
Connected successfully.
1 added the record
```

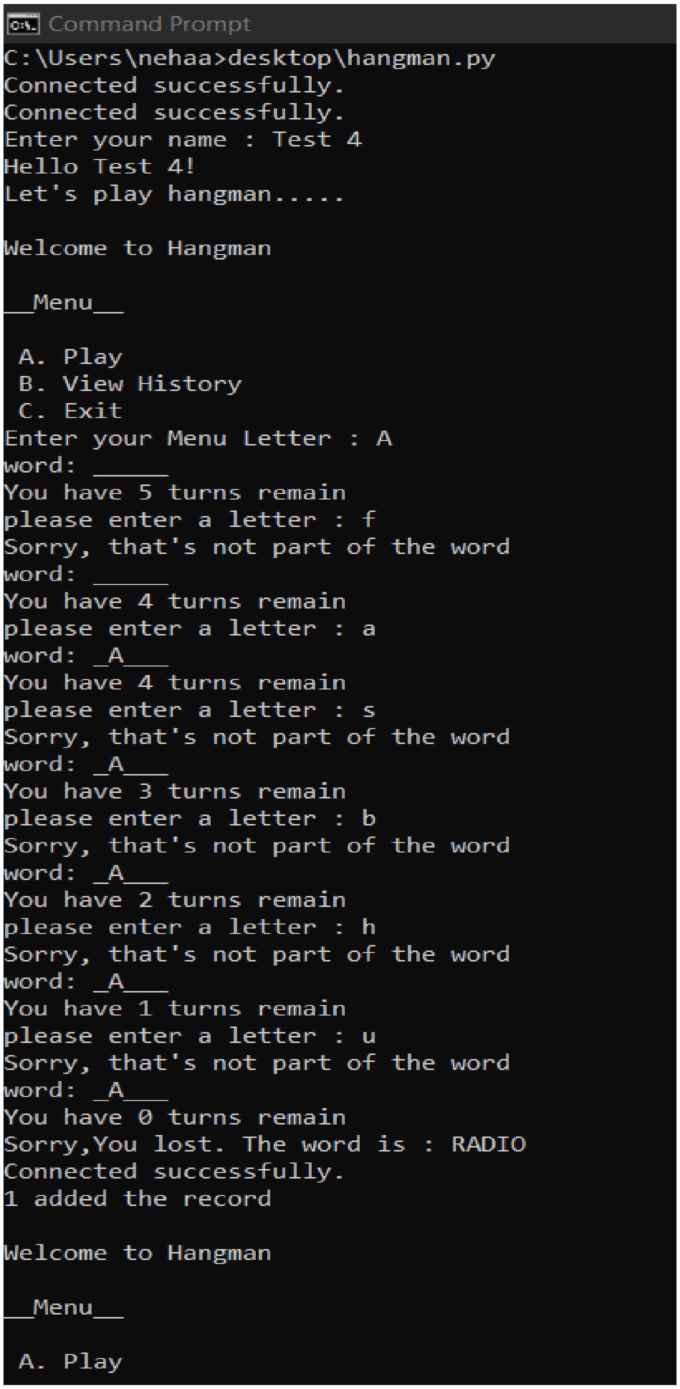
Figure 6



```
Welcome to Hangman

__Menu__

 A. Play
 B. View History
 C. Exit
Enter your Menu Letter : A
word: _____
You have 5 turns remain
please enter a letter : s
Sorry, that's not part of the word
word: _____
You have 4 turns remain
please enter a letter : f
Sorry, that's not part of the word
word: _____
You have 3 turns remain
please enter a letter : a
word: A____
You have 3 turns remain
please enter a letter : p
word: APP__
You have 3 turns remain
please enter a letter : l
word: APPL_
You have 3 turns remain
please enter a letter : e
Congratulations......You won!
The word is : APPLE
Connected successfully.
1 added the record

Welcome to Hangman

__Menu__

 A. Play
 B. View History
 C. Exit
Enter your Menu Letter : C
Game End!

C:\Users\nehaa>
```

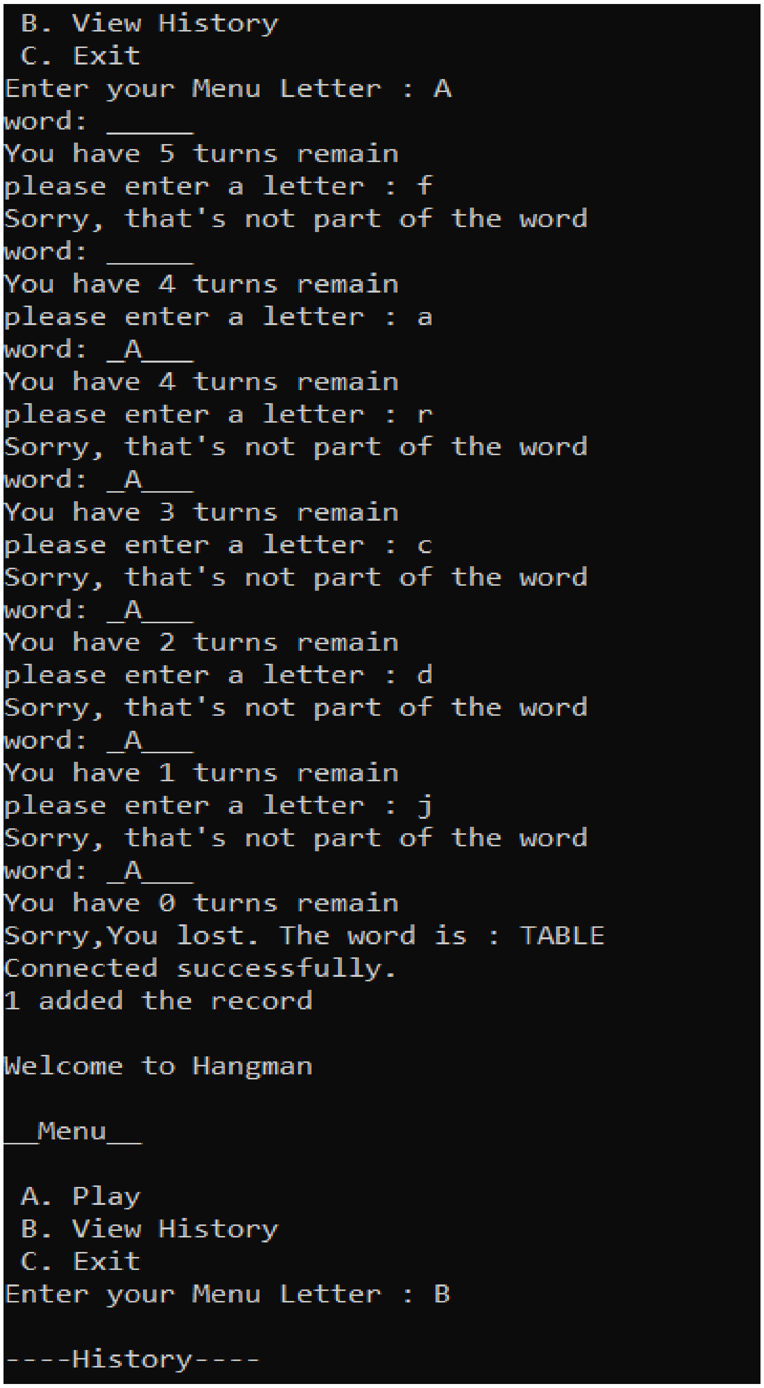
Figure 7

6.4 Test case 4



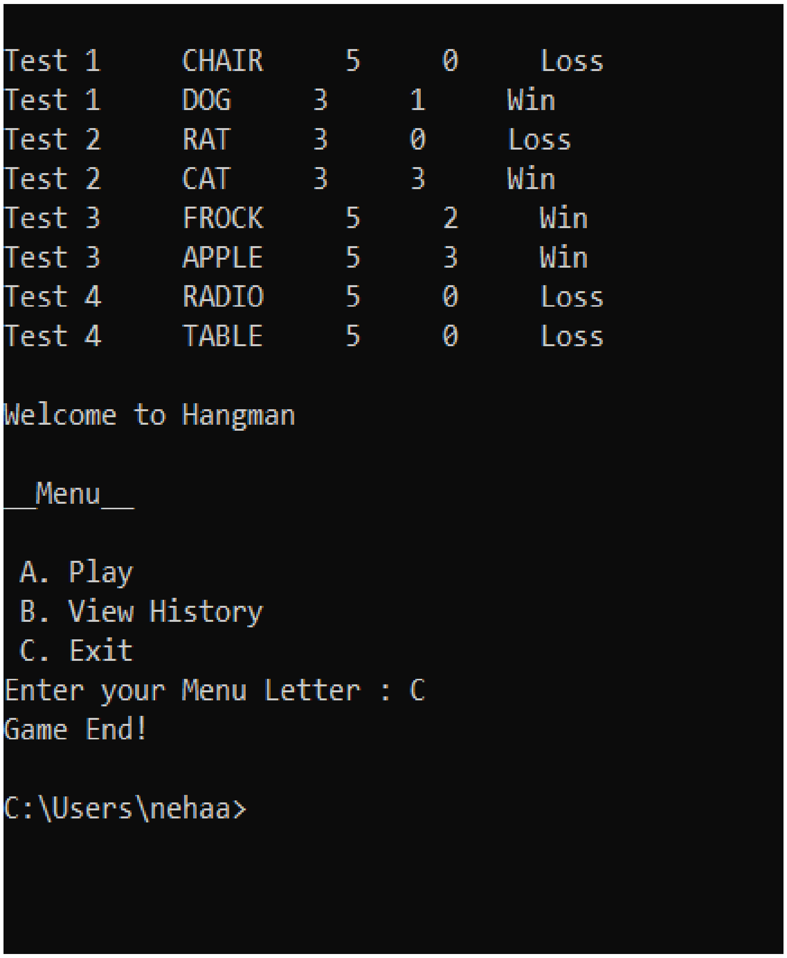
```
C:\Users\nehaa>desktop\hangman.py
Connected successfully.
Connected successfully.
Enter your name : Test 4
Hello Test 4!
Let's play hangman.....

Welcome to Hangman

__Menu__

 A. Play
 B. View History
 C. Exit
Enter your Menu Letter : A
word: _____
You have 5 turns remain
please enter a letter : f
Sorry, that's not part of the word
word: _____
You have 4 turns remain
please enter a letter : a
word: _A___
You have 4 turns remain
please enter a letter : s
Sorry, that's not part of the word
word: _A___
You have 3 turns remain
please enter a letter : b
Sorry, that's not part of the word
word: _A___
You have 2 turns remain
please enter a letter : h
Sorry, that's not part of the word
word: _A___
You have 1 turns remain
please enter a letter : u
Sorry, that's not part of the word
word: _A___
You have 0 turns remain
Sorry,You lost. The word is : RADIO
Connected successfully.
1 added the record

Welcome to Hangman

__Menu__

 A. Play
```

Figure 8



```
 B. View History
 C. Exit
Enter your Menu Letter : A
word: _____
You have 5 turns remain
please enter a letter : f
Sorry, that's not part of the word
word: _____
You have 4 turns remain
please enter a letter : a
word: _A___
You have 4 turns remain
please enter a letter : r
Sorry, that's not part of the word
word: _A___
You have 3 turns remain
please enter a letter : c
Sorry, that's not part of the word
word: _A___
You have 2 turns remain
please enter a letter : d
Sorry, that's not part of the word
word: _A___
You have 1 turns remain
please enter a letter : j
Sorry, that's not part of the word
word: _A___
You have 0 turns remain
Sorry,You lost. The word is : TABLE
Connected successfully.
1 added the record

Welcome to Hangman

__Menu__

 A. Play
 B. View History
 C. Exit
Enter your Menu Letter : B

----History----
```

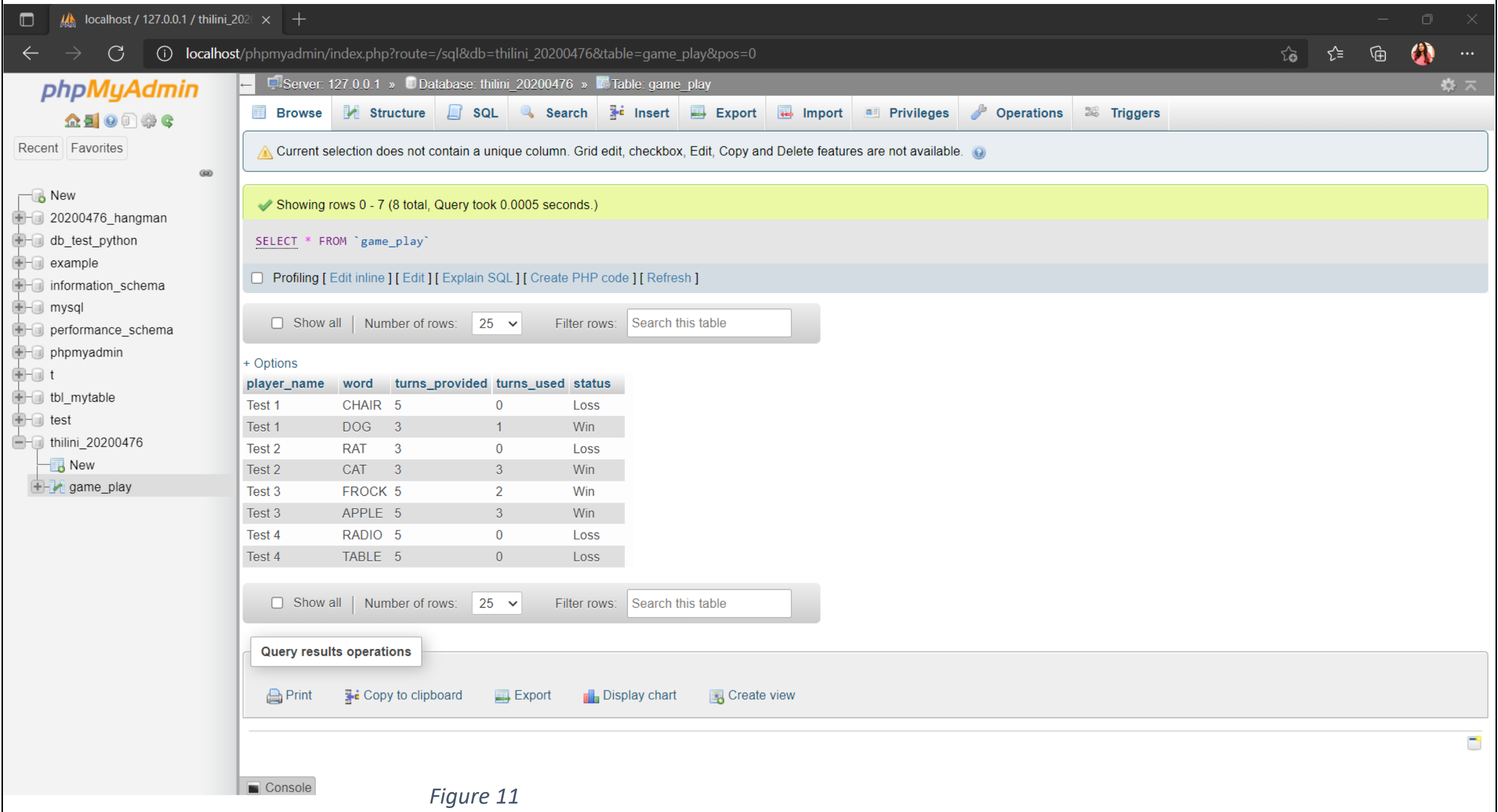
Figure 9



```
Test 1     CHAIR     5     0     Loss
Test 1     DOG     3     1     Win
Test 2     RAT     3     0     Loss
Test 2     CAT     3     3     Win
Test 3     FROCK     5     2     Win
Test 3     APPLE     5     3     Win
Test 4     RADIO     5     0     Loss
Test 4     TABLE     5     0     Loss

Welcome to Hangman

__Menu__

 A. Play
 B. View History
 C. Exit
Enter your Menu Letter : C
Game End!

C:\Users\nehaa>
```

Figure 10

6.5 Database test cases



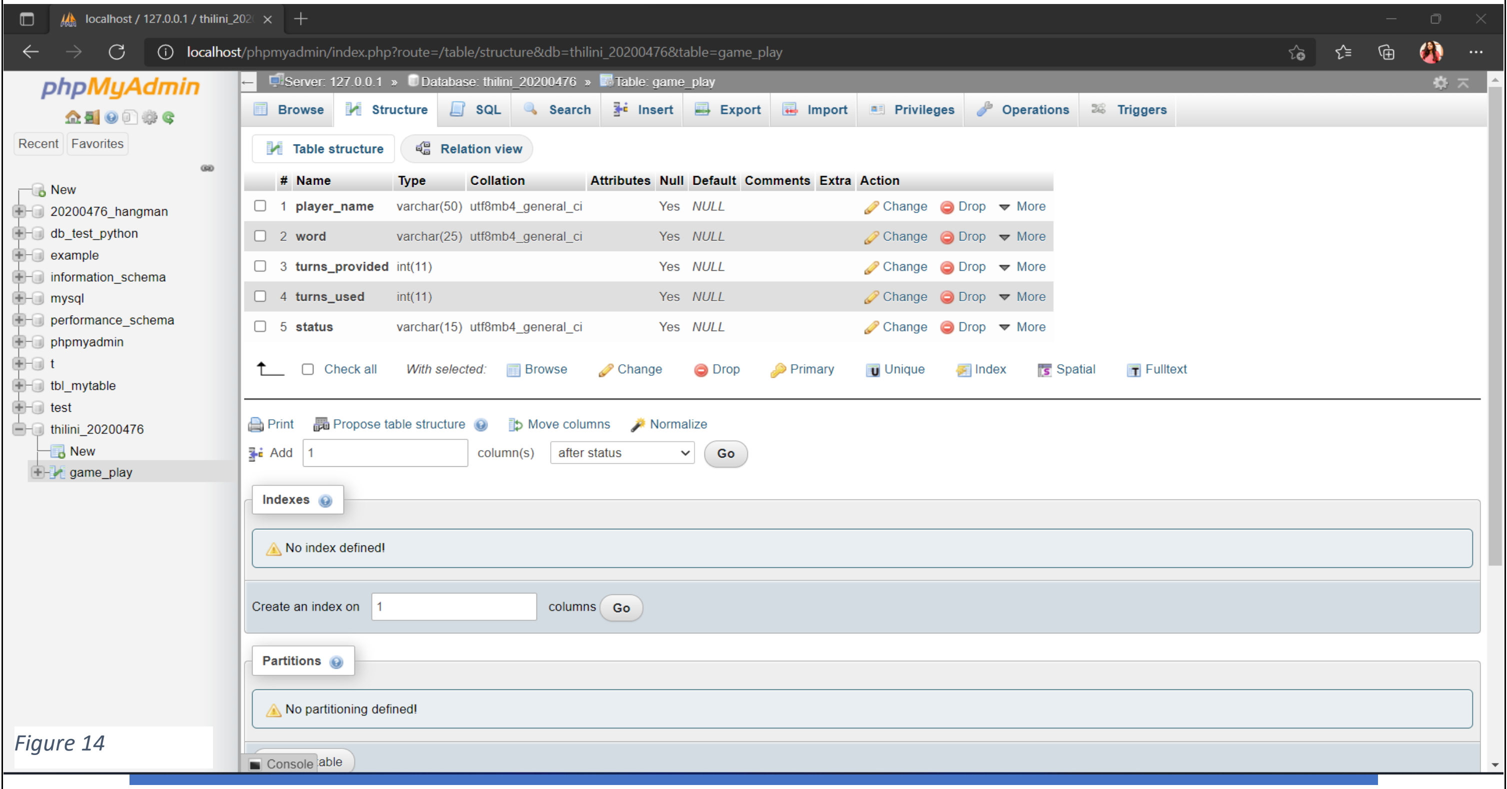
⚠ Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.

✓ Showing rows 0 - 7 (8 total, Query took 0.0005 seconds.)

```
SELECT * FROM `game_play`
```

player_name	word	turns_provided	turns_used	status
Test 1	CHAIR	5	0	Loss
Test 1	DOG	3	1	Win
Test 2	RAT	3	0	Loss
Test 2	CAT	3	3	Win
Test 3	FROCK	5	2	Win
Test 3	APPLE	5	3	Win
Test 4	RADIO	5	0	Loss
Test 4	TABLE	5	0	Loss

Figure 11



#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	player_name	varchar(50)	utf8mb4_general_ci		Yes	NULL			Change Drop More
2	word	varchar(25)	utf8mb4_general_ci		Yes	NULL			Change Drop More
3	turns_provided	int(11)			Yes	NULL			Change Drop More
4	turns_used	int(11)			Yes	NULL			Change Drop More
5	status	varchar(15)	utf8mb4_general_ci		Yes	NULL			Change Drop More

⚠ No index defined!

⚠ No partitioning defined!

Figure 14